

VECTORES

BIDIMENSIONAL Y MULTIDIMENSIONAL



*Ing. Tímotea Guadalupe Menjívar
Universidad Gerardo Barrios
Centro Regional Usulután
Facultad de Ciencia y Tecnología*



UNIVERSIDAD GERARDO BARRIOS

SEDE CENTRAL SAN MIGUEL /CENTRO REGIONAL USulután

Datos Generales	
Facultad	Ciencia y Tecnología
Asignatura	Fundamentos de programación
Docente	Ing. Timotea Guadalupe Menjivar
No. de Unidad	3
Contenido a desarrollar	VECTORES BIDIMENSIONALES Y MULTIDIMENSIONALES

1. INTRODUCCIÓN AL CONTENIDO

En el contenido de esta semana se examina la repetición (iteración) de sentencias en detalle y compara los bucles mientras y repetir, y para que nos ayudan a conocer cómo utilizar los vectores. Así mismo, la diferencia en cada una de ellas, lo que permite explicar las condiciones para su implementación, precauciones y reglas de uso de diseño de vectores. Compara los tres diferentes tipos de vectores vistos hasta el momento comparando su funcionamiento dependiendo de las situaciones, así como el concepto de vectores.

VECTORES BIDIMENSIONALES Y MULTIDIMENSIONALES

Vector Bidimensional

Un vector bidimensional es un conjunto de elementos, todos del mismo tipo, en el cual el orden de los componentes es significativo y en el que se necesita especificar dos subíndices para poder identificar cada elemento del vector. El vector bidimensional es un conjunto de vectores unidimensionales.

Los elementos de un vector bidimensional se referencian con dos subíndices:

- El primer subíndice se refiere a la fila.
- El segundo subíndice se refiere a la columna.
- En general se considera que un vector bidimensional comienza sus subíndices en 0 o en 1 según el lenguaje de programación. En pseudocódigo tomaremos como regla que inicia del 0, pero pueden tener límites seleccionados para el usuario durante la codificación del algoritmo.

Ejemplo: *creemos un vector bidimensional de 2x2:*

		COLUMNAS	
		0	1
FILAS	0	(0,0)	(0,1)
	1	(1,0)	(1,1)

En la tabla superior podemos ver que tenemos un vector o array de 2x2 es decir dos filas por 2 columnas. veamos un ejemplo en pseudocódigo.

Realice un vector que almacene el nombre y la edad de 4 personas.

Comenzamos por declarar variables y el array

```
1  Proceso ejm_bi
2      //definimos variables
3      definir dato,a como caracter;
4      definir i como entero;
5      //creamos el vector
6      Dimension dato(2,2);
```

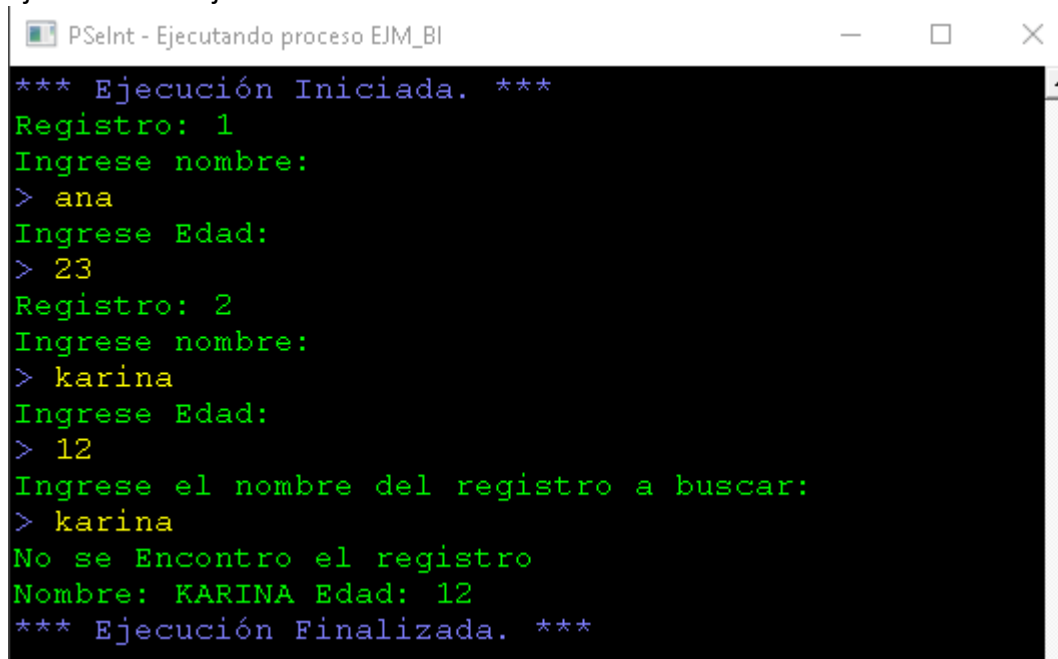
Dato(2,2) nos indica que es un vector o array de 2 filas por dos columnas.

Ahora bien vamos a realizar un ciclo para que nos permitirá hacer las iteraciones o repeticiones para pedir los datos.

Aca tenemos el código completo de este ejercicio:

```
1  Proceso ejm_bi
2      //definimos variables
3      definir dato,a como caracter;
4      definir i como entero;
5      //creamos el vector
6      Dimension dato(2,2);
7      Para i<-0 Hasta 1 Con Paso 1 Hacer
8          Escribir "Registro: ", i+1;
9          Escribir "Ingrese nombre: ";
10         Leer dato(i,0);
11         //hacemos que el nombre se guarde en mayusculas
12         dato(i,0)<-Mayusculas(dato(i,0));
13         Escribir "Ingrese Edad: ";
14         Leer dato(i,1);
15     FinPara
16     Escribir "Ingrese el nombre del registro a buscar: ";
17     Leer a;
18     a<-Mayusculas(a);
19     Para i<-0 Hasta 1 Con Paso 1 Hacer
20         Si dato(i,0)==a Entonces
21             Escribir "Nombre: ",dato(i,0)," Edad: ", dato(i,1);
22         SiNo
23             Escribir "No se Encontro el registro";
24         FinSi
25     FinPara
26 FinProceso
```

Ejecutamos el ejercicio



```
*** Ejecución Iniciada. ***
Registro: 1
Ingrese nombre:
> ana
Ingrese Edad:
> 23
Registro: 2
Ingrese nombre:
> karina
Ingrese Edad:
> 12
Ingrese el nombre del registro a buscar:
> karina
No se Encontro el registro
Nombre: KARINA Edad: 12
*** Ejecución Finalizada. ***
```

Ahora modifiquemos el ejercicio y hagamos que almacene 4 registros con los dos datos siempre para almacenar, convertiremos este array en un array de 4 filas por 2 columnas.

```
1  Proceso ejm_bi
2      //definimos variables
3      definir dato,a como caracter;
4      definir i como entero;
5      //creamos el vector
6      Dimension dato(4,2);
7      Para i<-0 Hasta 3 Con Paso 1 Hacer
8          Escribir "Registro: ", i+1;
9          Escribir "Ingrese nombre: ";
10         //La posición 0 de dato(i,0) almacena el nombre
11         Leer dato(i,0);
12         //hacemos que el nombre se guarde en mayusculas
13         dato(i,0)<-Mayusculas(dato(i,0));
14         Escribir "Ingrese Edad: ";
15         //La posición 1 de dato(i,1) almacena el edad
16         Leer dato(i,1);
17     FinPara
18     //Realiza la busqueda por nombre
19     Escribir "Ingrese el nombre del registro a buscar: ";
20     Leer a;
21     //El nombre lo convierte a mayúsculas
22     a<-Mayusculas(a);
```

Realicemos el ciclo de la búsqueda:

```
24     Para i<-0 Hasta 3 Con Paso 1 Hacer
25         Si dato(i,0)==a Entonces
26             Escribir "Nombre: ",dato(i,0)," Edad: ", dato(i,1);
27         SiNo
28             Escribir "No se Encontro el registro";
29         FinSi
30     FinPara
31 FinProceso
```

Al ejecutar el ejercicio podremos ver lo siguiente:

```
PSInt - Ejecutando proceso EJM_BI
Registro: 1
Ingrese nombre:
> ana
Ingrese Edad:
> 23
Registro: 2
Ingrese nombre:
> luis
Ingrese Edad:
> 12
Registro: 3
Ingrese nombre:
> jorge
Ingrese Edad:
> 34
Registro: 4
Ingrese nombre:
> maron
Ingrese Edad:
> 22
Ingrese el nombre del registro a buscar:
> jorge
No se Encontro el registro
No se Encontro el registro
Nombre: JORGE Edad: 34
No se Encontro el registro
*** Ejecución Finalizada. ***
```

Sigamos observadores veremos que este ciclo de búsqueda nos muestra que en la posición 2 del array (tomando en cuenta que inicia de 0) se almacena el registro de Jorge.

```
*** Ejecución Iniciada. ***
Registro: 1
Ingrese nombre:
> maria
Ingrese Edad:
> 23
Registro: 2
Ingrese nombre:
> maria
Ingrese Edad:
> 14
Registro: 3
Ingrese nombre:
> isolina
Ingrese Edad:
> 34
Registro: 4
Ingrese nombre:
> leonel
Ingrese Edad:
> 2
Ingrese el nombre del registro a buscar:
> maria
Nombre: MARIA Edad: 23
Nombre: MARIA Edad: 14
No se Encontro el registro
No se Encontro el registro
*** Ejecución Finalizada. ***
```

Vector Multidimensional

Se caracteriza por tener x números de filas como de columnas.

Por ejemplo necesitamos un vector que almacén el nombre, edad, genero y número de teléfono de 3 personas. Veamos

```
1  Proceso ejm_bi
2      //definimos variables
3      definir dato,a como caracter;
4      definir i como entero;
5      //creamos el vector
6      Dimension dato(3,4);
7      Para i<-0 Hasta 2 Con Paso 1 Hacer
8          Escribir "Registro: ", i+1;
9          Escribir "Ingrese nombre: ";
10         //La posición 0 de dato(i,0) almacena el nombre
11         Leer dato(i,0);
12         //hacemos que el nombre se guarde en mayusculas
13         dato(i,0)<-Mayusculas(dato(i,0));
14         Escribir "Ingrese Edad: ";
15         //La posición 1 de dato(i,1) almacena el edad
16         Leer dato(i,1);
17         Escribir "Ingrese Genero: ";
18         //La posición 1 de dato(i,2) almacena el genero
19         Leer dato(i,1);
20         Escribir "Ingrese Número de telefono: ";
21         //La posición 1 de dato(i,3) almacena el numero de tel.
22         Leer dato(i,1);
23     FinPara
24     //Realiza la busqueda por nombre
25     Escribir "Ingrese el nombre del registro a buscar: ";
26     Leer a;
27     //El nombre lo convierte a mayúsculas
28     a<-Mayusculas(a);
29
30     Para i<-0 Hasta 2 Con Paso 1 Hacer
31         Si dato(i,0)==a Entonces
32             Escribir "REGISTRO: ",i+1;
33             Escribir "Nombre: ",dato(i,0)," Edad: ", dato(i,1);
34             Escribir "Genero: ",dato(i,2)," Telefono: ", dato(i,3);
35
36         SiNo
37             Escribir "REGISTRO: ",i+1;
38             Escribir "No se Encontro el registro";
39         FinSi
40     FinPara
41 FinProceso
```

Veamos otro ejercicio:

Diseñar el algoritmo correspondiente a un programa, que:

- Crea una array de longitud 5x5 y nombre 'matriz'.
- Carga la tabla con valores numéricos enteros.
- Suma todos los elementos de cada fila y todos los elementos de cada columna

Visualizando los resultados en pantalla.

```
1  Proceso tabla
2      Definir matriz Como Entero;
3      Dimension matriz[5,5];
4      Definir fila,col como Entero;
5      Definir num_filas, num_cols Como Entero;
6      Definir suma como Entero;
7      num_filas<-5;
8      num_cols<-5;
9      Para fila<-0 hasta num_filas-1 Hacer
10         Para col<-0 hasta num_cols-1 Hacer
11             Escribir Sin Saltar "Introduce el número de la fila ",fila+1," y columna ",col+1,":";
12             Leer matriz[fila,col];
13         FinPara
14     FinPara
15     //Suma las filas
16     Para fila<-0 hasta num_filas-1 Hacer
17         suma<-0;
18         Para col<-0 hasta num_cols-1 Hacer
19             suma<-suma+ matriz[fila,col];
20         FinPara
21         Escribir "La suma de los elemento de la fila ",fila+1," es ",suma;
22     FinPara
23     //Suma las columnas
24     Para col<-0 hasta num_cols-1 Hacer
25         suma<-0;
26         Para fila<-0 hasta num_filas-1 Hacer
27             suma<-suma+ matriz[fila,col];
28         FinPara
29         Escribir "La suma de los elemento de la columna ",col+1," es ",suma;
30     FinPara
31 FinProceso
```

Ejercicio 3

Diseñar el algoritmo correspondiente a un programa, que:

Crea un array de longitud 5x15 y nombre 'marco'.

Carga la tabla con dos únicos valores 0 y 1, donde el valor uno ocupará las posiciones o elementos que delimitan la tabla, es decir, las más externas, mientras que el resto de los elementos contendrán el valor 0. Visualiza el contenido de la matriz en pantalla.


```
1  Proceso LadoExterno
2      Definir matriz Como Entero;
3      Dimension matriz[5,15];
4      Definir num_filas,num_cols como Entero;
5      Definir fila,col como Entero;
6      num_filas<-5;
7      num_cols<-15;
8      Para fila<-0 hasta num_filas-1 Hacer
9          Para col<-0 hasta num_cols-1 Hacer
10             //Si estoy en el extremo izquierdo, derecho, superior o inferior
11             Si fila=0 o fila=num_filas-1 o col=0 o col= num_cols-1 Entonces
12                 //Inicializo a 1
13                 matriz[fila,col]<-1;
14             SiNo//SiNo inicializo a 0
15                 matriz[fila,col]<-0;
16             FinSi
17         FinPara
18     FinPara
19     //Recorro para mostrar la tabla.
20     Para fila<-0 hasta num_filas-1 Hacer
21         Para col<-0 hasta num_cols-1 Hacer
22             Escribir Sin Saltar matriz[fila,col];
23         FinPara
24         Escribir " ";
25     FinPara
26 FinProceso
```

Practica de ejercicios

INDICACIONES: A continuación, se presenta una serie de ejercicios se solicita desarrollarlos de manera ordenada y dejando constancia de las operaciones. Resultados idénticos anulara la nota del ejercicio. La fecha de entrega de la guía será en la entrega del portafolio adjuntar los ejercicios. Realizar cada uno de los ejercicios haciendo uso de vectores. Copias idénticas entre equipos anula los ejercicios.

1. Escriba un programa que imprima la planilla de pago de 20 empleado. Los datos de salida son: nombre del empleado, sueldo bruto, AFP, seguro social, renta, descuentos, anticipos, sueldo neto. Luego que muestre los datos de empleados en orden alfabético.
2. Un programa que diga si los años comprendidos entre un rango establecido por el usuario son bisiesto o no en base a la siguiente afirmación:
"el año bisiesto requiere que las dos últimas cifras del año fueran divisibles por 4, pero que, además, el año no terminara en dos ceros; en este caso solo se consideraría bisiesto si fuera divisible por 400".
3. Realice un programa en P-seint usando el ciclo mientras y vectores que guarde los datos de x cantidad de libros. Los datos a almacenar son: título, autor, nombre, año de edición, código del libro, luego que genere los registros de menor a mayor. El programa debe terminar cuando el usuario escriba la palabra escribus.
4. Realice un programa que permita ingresar un listado de 3 notas, a 50 estudiantes de la carrera de ingeniería en sistemas ciclo I, las notas corresponden a laboratorio 1 y 2 que equivalen al 30% cada una, así mismo al parcial final.
También debe generar:
 1. El nombre del estudiante que tenga la nota máxima
 2. El nombre del estudiante que tenga la nota mínima.
 3. El listado de estudiantes cuyas notas sean mayores o iguales a 7.
 4. El porcentaje de hombres aprobados y reprobados
 5. El porcentaje de mujeres aprobadas y reprobados

RECURSOS COMPLEMENTARIOS

Recurso	Título	Cita referencial
Libro	<i>Fundamentos de programación 4ta Edición</i>	Aguilar, L. J. (2008). <i>Fundamentos de programación 4ta Edición</i> . Madrid: MCGRAW-HILL/INTERAMERICANA DE ESPAÑA, S. A. U.

ACTIVIDAD DE EVALUACIÓN DE LA SEMANA 1

Nombre de la Actividad	Evaluación en Línea (actividad formativa)
Tipo de Actividad	Cuestionario
Tipo de Participación	Individual
Instrucciones para la actividad	<p>Después de haber dado lectura a los contenidos de esta Unidad, ingrese a: Unidad 3- Actividad 1- Evaluación en Línea. La evaluación consta de preguntas y ejercicios desarrollados en Pseint de los cuales se le presentaran 5 de forma aleatoria. Para lograr el éxito en esta evaluación, deberá leer detenidamente cada uno de los ejercicios y preguntas, debe seleccionar la respuesta correcta.</p> <p>Recuerde que la evaluación solo dura 15 minutos y usted es responsable de la distribución de tiempo en cada pregunta. La evaluación estará disponible desde el viernes 22 de mayo hasta el día domingo.</p>
Fecha de Entrega	<ul style="list-style-type: none"> La fecha límite de participación será el domingo al final de la semana 4 a las 11: 55 p.m.

Evaluación dentro de 15 días.

ACTIVIDAD DE EVALUACIÓN DE LA SEMANA 2	
Nombre de la Actividad	Evaluando Conocimientos (Evaluativa)
Tipo de Actividad	Cuestionario
Tipo de Participación	Individual
Instrucciones para la actividad	<p>Después de haber dado lectura a los contenidos de esta Unidad, ingrese a: Unidad 3- Actividad 2- Evaluando Conocimientos.</p> <p>La evaluación consta de preguntas y ejercicios desarrollados en Pseint de los cuales se le presentaran 5 de forma aleatoria. Para lograr el éxito en esta evaluación, deberá leer detenidamente cada uno de los ejercicios y preguntas, debe seleccionar la respuesta correcta.</p> <p>Recuerde que la evaluación solo dura 15 minutos y usted es responsable de la distribución de tiempo en cada pregunta. La evaluación estará disponible desde el miércoles 27 de mayo hasta el día domingo.</p>
Fecha de Entrega	<ul style="list-style-type: none">La fecha límite de participación será el domingo al final de la semana 4 a las 11: 55 p.m.